Modeling The Joint Battlespace Infosphere

Robert G. Hillman
James P. Hanna
Martin J. Walter

Air Force Research Laboratory
Information Directorate
Advanced Computer Architectures Branch
26 Electronics Parkway
Rome, NY 13441-4514
(315) 330-4029

Abstract

The Air Force is developing a concept known as the Joint Battlespace Infosphere (JBI) in order to achieve information superiority. The JBI builds on the Global Information Grid and will move the state-of-the-art in information management into information-centric warfare. JBI will address more than five million information objects, thousands of users and provide scalability challenges for currently available information management hardware and software systems and drive design requirements to create systems that are more capable.

JBI concepts will initially be created working with prototypes and will eventually culminate in fielded implementations. Important issues, such as bandwidth, connectivity, computational requirements and storage, information protection and assurance, must be addressed. Significant efforts will be required to implement JBI resources, such as network technologies and topologies, required to achieve JBI's stated objective to achieve warfare information superiority.

Key JBI resources are being modeled and simulated to identify, quantify, and resolve technology and topology issues influencing the prototyping, development, and deployment of an operational JBI. This simulation research will allow JBI developers to identify and mitigate programmatic risk early enough within the JBI seven-year development window to allow successful development and deployment of JBI. This paper will discuss the proof of concept assessment performed and the resulting development.

Introduction

Future DoD campaign strategies are being developed based on deployable distributed information management systems that are integral parts of the overall deployed footprint [1]. Combined, this integrated information system will provide assets and individual users with time-critical information required for their function during the crisis or conflict. This new deployable information enterprise will allow DoD forces to maintain information superiority over adversaries and react accordingly. The DoD will invest substantial resources over the next decade on the development of the distributed information enterprise infrastructure. To this end, the Air Force is developing a concept known as the Joint Battlespace Infosphere (JBI) [2] to

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **JUN 2001**	2. REPORT TYPE	3. DATES COVERED **00-00-2001 to 00-00-2001**
4. TITLE AND SUBTITLE **Modeling the Joint Battlespace Infosphere**		5a. CONTRACT NUMBER
		5b. GRANT NUMBER
		5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)		5d. PROJECT NUMBER
		5e. TASK NUMBER
		5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Air Force Research Laboratory,Information Directorate,26 Electronics Parkway,Rome,NY,13441-4514**		8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		10. SPONSOR/MONITOR’S ACRONYM(S)
		11. SPONSOR/MONITOR’S REPORT NUMBER(S)
12. DISTRIBUTION/AVAILABILITY STATEMENT **Approved for public release; distribution unlimited**		
13. SUPPLEMENTARY NOTES **The original document contains color images.**		
14. ABSTRACT		
15. SUBJECT TERMS		

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**		**8**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

achieve information superiority. The other Services are developing similar concepts, such as the Army's "Digitized Battlefield" [3] and the Navy's "Network-Centric Warfare" [4]. These service system-of-systems concepts expand to include Joint and Coalition Task Force deployment systems. These systems will move the state-of-the-art information management from the current network-centric warfare designs into information-centric warfare, thus providing raw and fused information to end users at presently unachievable quality and rates.

No single information architecture will be established to meet the operational requirements. Numerous architectures will be developed and established; the overall requirement that all implementations interact seamlessly.

An efficient commercial quality simulation environment is required to support the development of complex deployable information systems. This simulation research is establishing the necessary capability allowing developers to identify and mitigate programmatic risk early enough within the development to allow successful development and deployment of the associated systems. The simulation framework will be capable of assessing deployment aspects such as security, quality of service, and fault tolerance. This development leverages previous DoD investments in event level simulation by building a generalized information architect simulator on top of SPEEDES [5,6].

The modeling framework will simulate the enterprise performance at the information level to identify, quantify, and resolve protocols, processes, and core functions influencing the prototyping, development, scalability and deployment of an operational enterprise. Simulations require "Challenge Problem" class resources to address more than five million information objects and hundreds of thousands of clients comprising a future information based force structure.

The design of a generalized simulation engine to support the modeling of a diverse set of distributed information systems posed considerable technical challenges. The conceptual framework, as seen in Figure 1, provides the user with the ability to separate the specification of nodal functionality from physical topology and communication protocols.

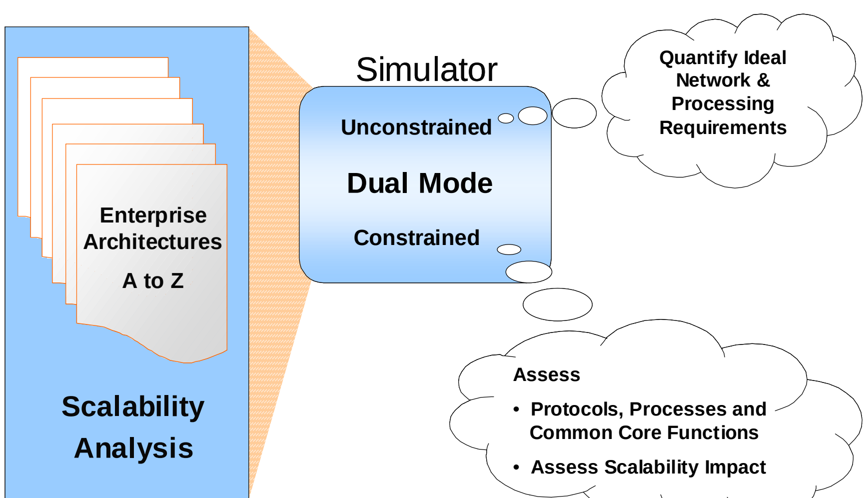

Figure 1. Conceptual Framework

Further, the framework provides the flexibility necessary to specify a broad set of nodal functionalities and physical topologies. The user can also specify operational scenarios necessary to drive the simulation and validate the model. The user is free from the necessity of writing code by defining model definitions for the information system enterprise in a declarative manner using configuration files.

Technical Approach

The goal of our framework design is to provide an underlying infrastructure and simulation engine that will allow users to model any information system enterprise in a declarative manner.

Through configuration files, the user specifies the system functionality, topology, and operational scenarios. The simulator reads these configuration files, builds the appropriate information system enterprise, and exercises the operational scenarios against the model. This technical approach is illustrated in Figure 2. The user is freed to focus on the design aspects rather than writing code to simulate and analyze the envisioned enterprise.

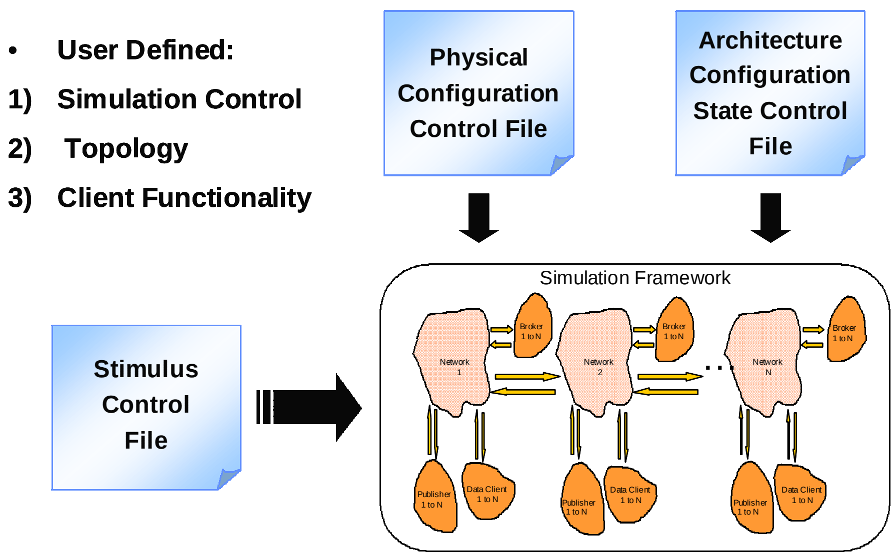

Figure 2. Technical Approach

The development was based on a nine-month study where prototypes of key aspects of the simulation framework were designed, coded, and evaluated. This evaluation phase was used to define the approach for the simulator design based on the lessons learned. The proof of concept coding was performed in the multiple environments; VHDL, JAVA and SPEEDES. This proof of concept phase reduces the project risk by verifying approaches and identifying the critical performance issues needing alternate solutions. This prototyping resulted in approximately 1200 lines of SPEEDES code written in C++.

A preliminary enterprise model exhibiting minimal behavior was used to validate the design approach for constructing an enterprise hierarchy. This model established a reference hierarchy between clients, providing some product (or functionality) to the enterprise and those that

subscribe to (or consume) the information. The event driven communication protocol (how messages will be transmitted and received) between enterprise clients was used to evaluate a structured network model and communications protocol. A design review was performed to evaluate both the development concept for the simulation engine as well as simulation performance. A comparative analysis was made to assess the simulation and identify the strengths, weaknesses, and applicability of different approaches for the architecture analysis.

The SPEEDES-based parallel processing framework was selected as our target environment since it provides an ideal foundation for the development of a generalized modeling tool to support simulation of distributed information systems on a multiprocessor platform. The SPEEDES simulation framework provides processing power not available from a single processor. Applications that make use of SPEEDES are typically time-constrained; too many events to process in a limited amount of time. SPEEDES allows the simulation builder to perform optimistic parallel processing on high performance computers, networks of workstations, or combinations of networked computers and HPC platforms.

Simulations were performed utilizing the preliminary architecture model. It was interesting to observe that with just over 500 clients, one million objects were processed and at 1000 clients 3 million objects were processed during initialization. The performance analysis, as seen in Figure 3, identified a major performance bottleneck in the network model used to connect the functional subsystems.

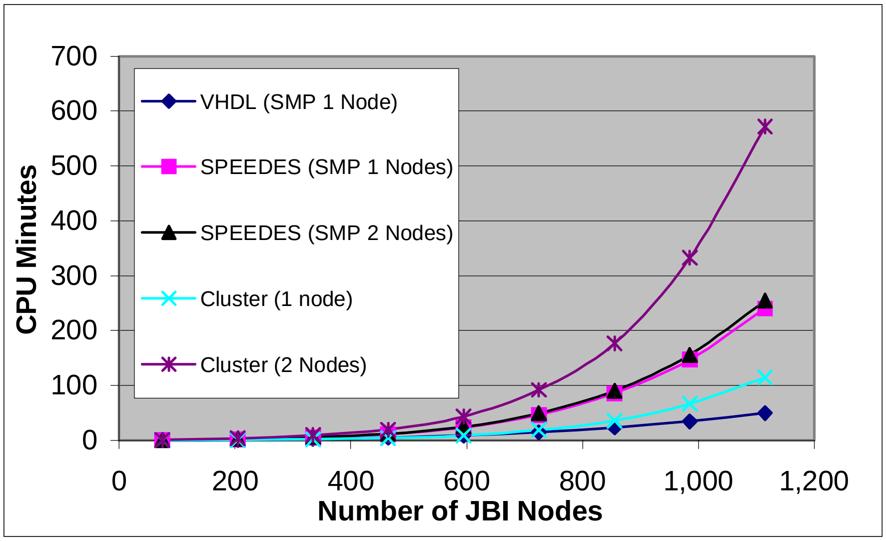

Figure 3. Performance Analysis

One thousand clients were simulated on a single CPU requiring 35 minutes to execute. A distributed simulation was performed on two nodes executing in 400 CPU minutes. As a result, a new prototype network model was coded utilizing a distributed approach resulting in orders of magnitude improvement for the distributed simulations. The resulting simulation employing up to 10 processor nodes to emulate 1500 clients executed in under 3 minutes. Simulations were expected to be highly event driven rather than CPU intense and the results confirmed that the event streams are definitely the overall performance-driving factor.

In order to further improve the HPC resource utilization, we evaluated several clustering algorithms to insure that clients with high event interactions would be constructed on common HPC nodes. As a result, the development optimization techniques have been evaluated for the simulation framework and the results are shown in figure 4. These optimization techniques were developed to balance the event distributions based on the simulation model.

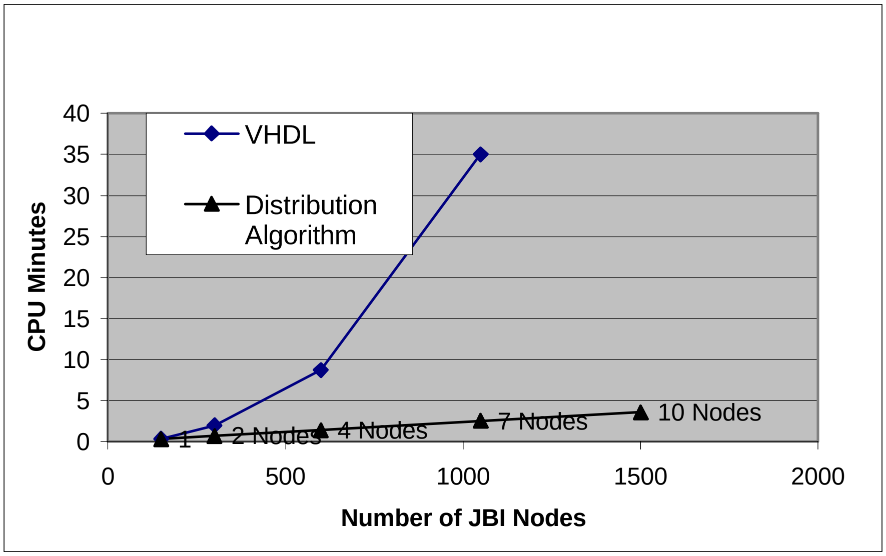

Figure 4. Optimized Distribution

Requirements Collection and Definition

Service implementations for the enterprise architectures will initially be created working with prototypes and eventually culminate in simulation of fielded implementations. The information enterprises will provide scalability challenges for currently available information management hardware and software systems and design requirements to create more capable systems. Each prototype and implementation will require numerous resources, including physical network implementations, as a basis to support design, development, and operational efforts. Important issues such as computational requirements, storage, information protection and assurance, bandwidth, and connectivity must be addressed. Significant effort will be required to assess and evaluate these potential resource requirements. The distributed information enterprise must be simulated based on the operational behaviors of the members in conjunction with the physical constraints of the resources.

We are collecting and identifying all simulation framework requirements and feeding these requirements into the development of the core simulator.

1. Functional behavioral requirements will be extracted from prototype enterprise implementations that are being developed.
2. State machine representations of the behavioral functions that are to be imbedded in the information clients will be developed.
3. Physical resource characteristics will be identified for the underlying computer and network topologies from which model instantiations can be developed.

Client Functional Capabilities

Distributed information systems consist of a collection of autonomous clients interconnected via a network. Each client provides some product (or functionality) to the enterprise, and perhaps subscribes to (or consumes) one or more enterprise products. The specification of system topology is rather straightforward. The specification of individual client functionality, however, is not so simple. A given enterprise model may consist of thousands of clients distributed across tens or even hundreds of network domains. Client functionality cannot be achieved through static parameter instantiation; it requires a more sophisticated mechanism.

Our approach is to provide the mechanism to specify the behavior of each client as a state table. This state table will be a local client data structure built during the initialization phase of the simulation and will be based on an XML configuration file. The states enumerated in the state table will be callable methods that define atomic actions. Each state in the table may also consist of pre- and post- conditionals, as well as parameters necessary to fully define the atomic action desired. The order and aggregation of a given set of atomic actions will define the functionality of the client. Each client will iterate through its state table when it receives an activation event. The state pre-conditional may be used to determine whether a given state is active during the given cycle. The state post-conditional may be used to trigger events or set pre-conditionals of subsequent states. This approach will allow a broad set of functionality to be specified for clients.

All atomic actions are defined as SPEEDES public C++ class methods. Data types and objects, including conditionals, are defined using C++ container classes that are inherited by the atomic actions class. There is a one-to-one mapping between the elements specified in the XML configuration file and the methods defined in the atomic actions class. Instantiation of an atomic action object within a client creates all necessary data storage and provides visibility to all atomic action methods. An initialization method will read an XML configuration file and initialize the state table and other local data after construction of the atomic action object. A second XLM configuration file will be used as a stimulus to drive the operational scenarios.

One technical challenge to this approach is the formation of a complete enough set of atomic actions that would allow for the specification of a diverse set of functionality. This goal may not be completely achievable. A library of actions will be developed based on a selected set of core capabilities that must be supported by the Air Force JBI initiative and selected enterprise applications. Several implementations of the JBI publish-subscribe scenario currently exist and will provide a good starting point for atomic action specification in the initial year.

A key aspect to this modeling approach is its extensibility. Additional methods can be added to the atomic action class definition as needed and the XML configuration file can reflect these additions to provide access to new atomic actions. Unfortunately this approach requires the user to develop C++ code and integrate this code into the simulator. A fallback approach would be to develop a user Application Programming Interface (API) for extending the functionality of the atomic actions class. Both approaches will be analyzed to determine the optimal solution.

Object Level Network Model

The interconnection network defines the bandwidth (or capacity) and latencies (or delay) of the connections between clients. Creating collections of clients with specific processing capabilities and networking characteristics can be achieved through instantiation with static parameters. Multiple network domains are modeled by associating sets of node clients with a given network domain and defining bandwidth and latency as a function of domain locality. In this approach, aggregating multiple network domains is achieved simply by partitioning the enterprise by domain association. Again, this can be achieved through instantiation with static parameters using an XML configuration file.

Modeling the information flow, among the clients in an enterprise containing many thousands of processing members in order to analyze associated enterprise scaling issues, must be performed utilizing model abstraction. The abstract models of the network fabric and the data packages will be constructed. This model abstraction, which will be part of the simulator framework, will represent the network described by high-level abstract characteristics, such as the latency and bandwidth of an equivalent link between any two nodes on the network. The functionality of the network and its time varying usage will be integrated with the simulation so that the effects of load, capacity, and interconnection can be adequately analyzed.

The simulated behavior of these abstract models must relate to the real world implementations. Therefore, these models must be calibrated and verified against a known reference, OPNET [7]. OPNET is the industry de-facto standard for emulating detailed network behavior. The detail this network simulator provides can validate the scaling issues related to our abstract network model.

To calibrate and verify the abstracted network models developed for the JBISIM framework, network combinations with client clusters will be modeled with both OPNET and JBISIM. These simulations will establish the reference latency and bandwidth between processing clients in the network for use in the JBISIM Simulator. The abstract JBISIM models will be refined by back annotation using the OPNET simulations for identical networks with dozens of processing clients and a few subnets. Having developed and calibrated an abstract model for node-to-node latency and bandwidth on small and medium sized network segments, scaled simulations will be performed with thousands of nodes to validate the JBISIM network model. OPNET simulations of the same network configuration and data flow will be made. This iterative analysis process will calibrate the JBISIM network modeling, and verify its applicability to scaling issues, identifying limitations regarding precision and accuracy of the simulation. Discrepancies between the simulation results will be analyzed and used to adjust the precision and scalability issues of the simulator. Some variations are expected to remain but the goal is to develop the mapping algorithms so that a consistency within 10% is maintained.

Reference Architecture

All testing and evaluations of the simulation framework during the development process will initially be targeted at a reference enterprise architecture. This reference enterprise architecture will be developed based on system requirements collected. The intention is to develop a

reference enterprise that will aggregate the behavioral requirements into a single test reference. This reference enterprise will be used to evaluate scalability and performance during the development process.

The reference enterprise will also be used in the demonstration phase. As the common model, it will be utilized for contrasting the scalability and performance results of the four target HPC systems. The reference architecture will be simulated on each system and statistics related to scalability and performance documented. Performance improvements addressing interconnect latency related to the pre- and post- optimization for SPEEDES will also be documented.

Conclusion

The proposed simulation capability will give the Air Force JBI team the ability to model key JBI related resources, identify, quantify, and resolve technology and topology issues influencing the prototyping, development, and deployment of the operational JBI enterprise. This simulation research will allow JBI developers to identify and mitigate programmatic risk within the JBI seven-year development schedule and help in the successful development and deployment of JBI enterprise. It is anticipated the Air Force Research Laboratory, Information Directorate, will support this simulation framework throughout the development years and beyond. During this period the simulator will be provided to any US Government agency requesting the framework to support their mission requirements.

References:

[1] United States Air Force Scientific Advisory Board, "Air Force Command & Control - The Path Ahead", Summary Report (Volume 1), SAB-TR-00-01, December 2000

[2] United States Air Force Scientific Advisory Board, "Report on Building the Joint Battlespace Infosphere, Volumes 1 and 2," SAB-TR-99-02, December 1999

[3] Frankel, Michael S., "The 1994 Army Science Board Recommended Technical Architecture for the Digital Battlefield", Army RD&A Bulletin, December 1994

[4] Alberts, David S., John J. Garstka, and Frederick P. Stein, "Network Centric Warfare: The Face of Battle in the 21st Century", Washington, DC: National Defense University Press, 1999.

[5] J. Steinman, "Scalable Parallel and Distributed Military Simulations Using the SPEEDES Framework," Elecsim 95, 1995.

[6] Gary Blank, Jeff Steinman, , "Design and Implementation of the High Performance Computing RTI for the High-Level Architecture", Proceedings Simulation Interoperability Workshop, Spring 2000

[7] Irene Katzela, "Modeling and Simulating Communication Networks - A Hands-on Approach Using OPNET", Prentice Hall, ISBN 0-13-915737-9, 1998